3G-FARM

ソフトウェアマニュアル

目次

1 はじめに

このたびは 3G-FARM をお求めいただき、ありがとうございます。

3G-FARM は通信・Linux・省電力を融合させた通信ボードです。

通信
FOMA ユビキタスモジュール UM01-HW を搭載、FOMA の広い通信エリアと安定した通信が可能です。

Linux
Armadillo-420 を搭載、組み込み済の Linux で TCP/IP や USB などを容易に使用できます。

省電力
UM01-HW のスリープモードをフルに生かせる回路構成で電力消費を削減できます。

広い動作温度範囲
動作温度範囲は-20℃～+60℃までカバーしています。山間部や農場での使用が可能です。

■

本マニュアルは、3G-FARM のソフトウェア仕様や使用方法について書かれたものです。3G-FARM の機能を最大限引き出すために本マニュアルご活用いただければ幸いです。

説明の都合から一部の表現を 表 1-1のように省略名で記載しています。

表 1-1 名称の省略

省略名	正式名	備考
A420	Armadillo-420	ARM コア採用 CPU ボード
PIC	PIC24FJ128GA108	16bit PIC マイコン

2 注意事項

2.1 安全に関する注意事項

3G-FARM を安全にご使用いただくために、特に以下の点にご注意くださいますようお願いいたします。

本製品には一般電子機器用（ＯＡ機器・通信機器・計測機器・工作機械等）に製造された半導体部品を使用しておりますので、その誤作動や故障が直接生命を脅かしたり、身体・財産等に危害を及ぼす恐れのある装置（医療機器・交通機器・燃焼制御・安全装置等）に組み込んで使用しないでください。

また、半導体部品を使用した製品は、外来ノイズやサージにより誤作動したり故障したりする可能性がありますので、ご使用になる場合は万一誤作動、故障した場合においても生命・身体・財産等が侵害されることのないよう、装置としての安全設計（リミットスイッチやヒューズ・ブレーカ等の保護回路の設置、装置の多重化等）に万全を期されますようお願い申しあげます。

2.2 取り扱い上の注意事項

3G-FARM や他のボードに恒久的なダメージをあたえないよう、取り扱い時には以下のような点にご注意ください。

- 電源の投入
  3G-FARM や他のボードに電源が入っている状態では絶対にコネクタやモジュール類の着脱を行わないでください。※Armadillo-420 の LAN、USB を除く。
- 静電気
  3G-FARM には CMOS デバイスを使用しておりますので、ご使用になるまでは帯電防止対策のされている、出荷時のパッケージ等にて保管してください。
- ラッチアップ
  電源および入出力からの過大なノイズやサージ、電源電圧の急激な変動等で使用している CMOS デバイスがラッチアップを起こす可能性があります。いったんラッチアップ状態となると、電源を切断しないかぎりこの状態が維持されるため、デバイスの破損につながることがあります。ノイズの影響を受けやすい入出力ラインには保護回路を入れることや、ノイズ源となる装置と共通の電源を使用しない等の対策をとることをお勧めします。

2.3 商標について

- FOMA、FOMA ユビキタスモジュールは株式会社エヌ･ティ･ティ･ドコモの登録商標です。
- Armadillo は株式会社アットマークテクノの登録商標です。
- その他本書に記載している会社名および商品名は、各社・各団体の商標または登録商標です。

2.4 ソフトウェア使用に関しての注意事項

本製品のために弊社から提供するソフトウェア（ドキュメント・サンプル等も含む）は、現状のまま（AS IS）提供されるものであり、特定の目的に適合することや、その信頼性、正確性を保証するものではありません。また、本製品の使用による結果についてもなんら保証するものではありません。

3 資料・参考文献

本マニュアル記載の内容を補完する資料・参考文献を以下に示します。

- FOMA ユビキタスモジュール(FOMA UM01-HW)取扱説明書
  FOMA ユビキタスモジュール(FOMA UM01-HW)組込みガイドライン
  UM01-HWモジュールについて説明されています。株式会社エヌ･ティ･ティ･ドコモのWEBサイトからダウンロードすることができます。
  ドコモビジネスオンライン(http://www.docomo.biz/)
  ドコモビジネスオンラインUMテクニカルサポート
  (http://www.docomo.biz/html/service/module/support/login.html)
  ※ドコモビジネスオンライン会員登録とUMテクニカルサポート規約への同意が必要です。

- Armadillo-400 シリーズハードウェアマニュアル
  Armadillo-400 シリーズソフトウェアマニュアル
  Armadillo 実践開発ガイド(第１部～第３部)
  搭載されているメインCPUのマニュアルです。株式会社アットマークテクノのWEBサイト(http://armadillo.atmark-techno.com/)からダウンロードすることができます。

- PIC24FJ256GB110 Family Data Sheet
  搭載されているサブ CPU のデーターシートです。マイクロチップ・テクノロジーのWEB サイト(http://www.microchip.com/)からダウンロードすることができます。また同サイトでは多数のアプリケーション例が公開されています。

- PICkit3 Programmer/Debugger User's Guide
  開発ツール PICkit3 のマニュアルです。マイクロチップ・テクノロジーWEB サイト(http://www.microchip.com/)からダウンロードすることができます。

- S-35390A マニュアル Rev3
  搭載されているカレンダ時計について説明されています。セイコーインスツル株式会社の WEB サイト(http://www.sii.co.jp/)からダウンロードすることができます。

- 3G-FARM ハードウエアマニュアル
  3G-FARM マニュアルのハードウェア編です。ハードウェア仕様が説明されています。

- 3G-FARM スタートアップマニュアル
  3G-FARM の開発環境を準備し、PPP 通信を行うまでの手順を説明しています。

4 アプリケーション開発環境

3G-FARM のアプリケーション開発には、以下のソフトウェアをご用意ください。

- VMware Player (仮想マシン)
  http://www.vmware.com/jp/products/desktop_virtualization/player/overview
- ATDE3(VMware 仮想イメージファイル) V20100309
  http://armadillo.atmark-techno.com/armadillo-420/downloads
- Hermit At Win32 (Armadill フラッシュメモリ書き換えツール)
  http://armadillo.atmark-techno.com/armadillo-420/downloads
- 通信ソフトウェア
  Teraterm などの使用可能な通信ソフトウェア
- Microcip MPLAB IDE (PIC 統合開発環境) ※
  http://www.microchip.com/
- Microcip MPLAB C30 (PIC C コンパイラ) ※
  http://www.microchip.com/

※ PIC 上で動作するアプリケーションの開発を行わない場合には使用しません。

PIC24FJ128GA108 内蔵フラッシュメモリに作成したアプリケーションを書き込むためには、ICSP(イン・サーキット・シリアル・プログラミング)に対応した書き込み器が必要です。マイクロチップ・テクノロジーの PICkit3 または PICkit2 などの書き込み器をご用意ください。

5 3G-FARM のソフトウェア概要

ここでは 3G-FARM のソフトウェア概要を簡単に説明します。

5.1 ブロック図

3G-FARM のブロック図を図 5-1 に示します。
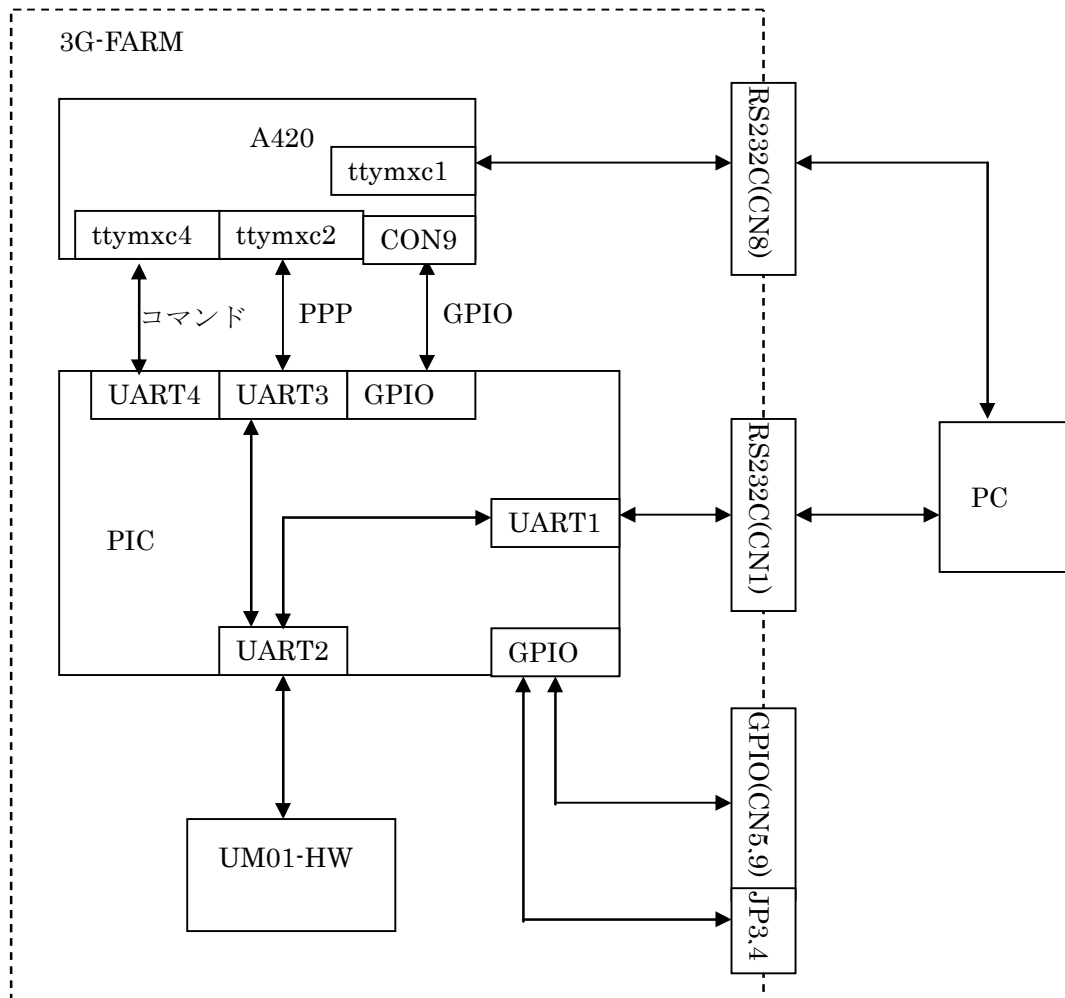
図 5-1 3G-FARM ブロック図

PIC の UART はリマッピング機能によりピンアサインを自由に変更することができますが、本マニュアルでは UART1 から 4 をこの図のとおり割り当てるものとして説明しています。出荷時に PIC に書き込まれている PIC 標準ソフトウェア(3GFARMMON)も、この設定となっています。

5.2 PIC と Armadillo-420 との接続

PIC と A420 の間は 2 チャンネルのシリアルポートと、2 ビットの GPIO で接続されています。信号については 3G-FARM ハードウェアマニュアル 6.4 項をご参照ください。PIC に出荷時書き込みされている標準ファームウェア 3GFARMMON では、PPP 通信は A420 の ttymcx2 を使い、ttymcx4 は PIC の動作制御コマンド(例えば UM01-HW の電源 ON/OFF、スリープ設定など)に使用します。また CON9_1 を UART 切替に使用しています。

5.3 出荷時の A420 カーネル

Armadillo-420 には、ttymxc2(UART3)のハードウェアフローコントロールを有効にしたカーネルが書き込まれています。カーネルコンフィギュレーションの変更方法は 3G-FARM スタートアップマニュアルをご参照ください。

5.4 出荷時の A420 ユーザランド

Armadillo-420 のユーザランドには Armadillo-420 の標準イメージの/root/に 3G-FARM を使用するためのサンプルシェルスクリプトを追加しています。サンプルシェルスクリプトについては9章をご参照ください。出荷時のユーザランドイメージは、マニュアルディスクに収録されています。

5.5出荷時の PIC ファームウェア

PIC には UM01-HW の制御や、Armaxdillo-420 と UM01-HW との通信を中継するための標準アプリケーション(3GFARMMON)が書き込まれています。使用方法は8章をご参照ください。オブジェクトファイル(3GFARMMON.HEX)は、マニュアルディスクに収録されています。

Armaillo-420 からの UM01-HW 操作、PIC の GPIO 操作等は 3GFARMMON を通して行うことができます。Armadillo-420 で動作させるアプリケーションが 3GFARMMON でサポートされている機能のみで実現できる場合は、PIC で動作するファームウェアを作成する必要はありません。

6 ソフトウェア

この章では、ソフトウェアに関する事項を説明します。CPU のレジスタ機能、UM01-HW の AT コマンド等については UM01-HW のハードウェアマニュアル・データシートをご参照ください。

6.1 Armadillo-420(メイン CPU)

6.1.1 開発環境

株式会社アットマークテクノの ATDE3 を使用します。詳しくは同社の Armadillo サイト(http://armadillo.atmark-techno.com/)をご参照ください。

6.1.2 UART

PIC 標準ファームウェア 3GFARMMON を使用する場合、Armadillo-420 の ttymxc2 を PPP 接続に、ttymxc4 を PIC や UM01-HW の制御に使用します。
ttymxc2 は CTS/RTS を使用したハードウェアフローコントロールを使用します。出荷時 Armadillo-420 に書き込み済みのカーネルではこの設定が有効になっていますが、カーネルをビルドして新たに作成する場合は、カーネルコンフィギュレーションで ttymxc2 のハードウェアフローコントロールを有効になるように設定してください。(3G-FARM スタートアップマニュアル 7 章をご参照ください。)

6.1.3 PPP

出荷時の Armadillo-420 カーネルは PPP 機能が有効になっており、ユーザランドには pppd や PPP の設定ファイル等が用意されていますので、3G-FARM ではプロバイダから提供された情報を設定ファイルに書き込むだけで PPP 通信を行うことができます。
PPP 通信に関連する設定ファイルの設定や、pppd の起動方法については 3G-FARM スタートアップマニュアル 6 章、8 章をご参照ください。

6.2 PIC (サブ CPU)

6.2.1 開発環境

PIC で動作するアプリケーションを作成する場合は、MPLAB IDE(開発環境)と MPLAB C30(C コンパイラ)を使用します。ソフトウェアやドキュメントはマイクロチップ・テクノロジーの WEB サイトから無償でダウンロードできます。
作成したアプリケーションを PIC 内蔵フラッシュメモリに書き込みする場合は、ICSP(イン・サーキット・シリアル・プログラミング)に対応した PICkit2/PICkit3 などの書き込み器が必要です。

6.2.2 UART

PIC には 4 チャンネルの UART があり、リマッピング機能によりピン割り当てを自由に変更できます。標準ファームウェア 3GFARMMON での各 UART の割り当てを表 6-1に示します。

表 6-1 PIC UART 番号

UART 番号	接続先
UART1	CN1(RS232C)
UART2	UM01-HW UART
UART3	A420(ttymxc2)
UART4	A420(ttymxc4)

UART 送受信端子の割り当ては、表 6-2のとおりです。

表 6-2 PIC UART ピンアサイン

UART 番号	TX	RX
UART1	RP19	RPI33
UART2	RP30	RPI36
UART3	RP10	RPI43
UART4	RP5	RP29

送受信以外のシリアルポート制御信号は、GPIO として制御してください。入出力レベル変換回路が接続されている端子の入出力設定は、各端子の入出力にあわせて設定してください。

表 6-2の設定に対応するプログラム例を図 6-1に示します。

図 6-1 UART リマッピング設定

```
#include <p24FJ128GA108.h>
#define UART1_TX  3
#define UART2_TX  5
#define UART3_TX  28
#define UART4_TX  30

#define Po_RP19   RPOR9bits.RP19R
#define Po_RP30   RPOR15bits.RP30R
#define Po_RP10   RPOR5bits.RP10R
#define Po_RP05   RPOR2bits.RP5R

#define UART1_RX  RPINR18bits.U1RXR
#define UART2_RX  RPINR19bits.U2RXR
#define UART3_RX  RPINR17bits.U3RXR
#define UART4_RX  RPINR27bits.U4RXR

Po_RP19 = UART1_TX;
UART1_RX = Pi_RPI33;
Po_RP30 = UART2_TX;
UART2_RX = Pi_RPI36;
Po_RP10 = UART3_TX;
UART3_RX = Pi_RPI43;
Po_RP05 = UART4_TX;
UART4_RX = Pi_RPI29;
```

6.2.3 UART 通信の中継

UM01-HW は PIC に接続されており、Armadillo-420 には直接接続されていないため、Armadiilo-420 から UM01-HW を使用する場合は、Armadillo-420 と UM01-HW 間の通信を中継するアプリケーションを PIC で動作させる必要があります。
PIC 標準ファームウェア 3GFARMMON では、Armadillo-420 の CON9_1 に 0 が出力されている場合 UM01-HW の送受信と RTS、CTS の信号を Armadillo-420 の ttymxc2 に中継しています。
なお Armadiilo-420 と PIC の間は ttymxc2 と ttymxc4 の 2 チャンネル UART で接続されていますが、フローコントロールを使用できるのは ttymxc2 のみです。

6.2.4 UM01-HW の電源制御

PIC のアプリケーションで UM01-HW の電源 ON、OFF を行う場合は、ライブラリの電源制御用関数(7.2節)を使用してください。これらの関数を使用しない場合は、FOMA ユビキタスモジュール(FOMA UM01-HW)組込みガイドラインに従い、正しい電源 ON/OFF シーケンスとなるようにプログラムを作成してください。
UM01-HW は電源起動後に UIM カードを参照して UM01-HW 内部の設定を自動的に行いますが、この間最大 30 秒程度は通信操作を行うことができません。AT コマンドを使用して UM01-HW が通信可能な状態となっているかを確認する必要があります。詳細については UM01-HW 組込ガイドラインをご参照ください。

6.2.5 JP3、JP4

JP3、JP4 はそれぞれ PIC の RF0、RD7 に接続されています。3GFARMMON では動作モードの変更に JP4 ジャンパを使用しています。PIC で動作するアプリケーションを作成する場合は、JP3、JP4 をアプリケーションで自由に使用できます。

6.2.6 I²C の接続制御

RTC は Armadillo-420 と PIC の両方に I²C で接続されています。Armadillo-420 から RTC にアクセスする場合は PIC の RC3 に 1 を出力してください。RC3 に 0 が出力されている場合は Armadillo-420 から RTC にはアクセスできません。
Armadillo-420 の電源をオフにする場合は、RC3 に 0 を設定してください。
３GFARMMON で Armadillo-420 の電源制御を行う場合、および提供されているライブラリの電源制御関数を使用する場合は、この端子の制御も同時に行われます。

6.2.7 CON9_1、CON9_2

Armadillo-420 の CON9_1 と CON9_2 はそれぞれ PIC の RF3、RF5 に接続されています。PIC 標準ファームウェア 3GFARMMON では CON9_1 を使用して PPP 用のシリアルチャンネルと、制御コマンド送受信用のシリアルチャンネルとの切替を行っています。
PIC で動作するアプリケーションを作成する場合は、この端子をアプリケーションで自由に使用できます。

6.3 UM01-HW(通信モジュール)

6.3.1 ソフトウェア更新機能の実装

UM01-HW を使用するアプリケーションには、UM01-HW のソフトウェア更新機能を組み込むことが必須とされています。3G-FARM ではソフトウェア更新機能を um01hw_softwareUpdate 関数としてライブラリに用意していますので、PIC のアプリケーションを作成する場合はこの関数を必ず組みこんでください。ソフトウェア更新関数については7.5.1項をご参照ください。
PIC の標準ファームウェア(3GFARMMON)を使用する場合は、Armadillo-420 からソフトウェア更新コマンドを使用してソフトウェア更新を実行することができます。ソフトウェア更新コマンドについては8.5.3項をご参照ください。

<u>ソフトウェア更新における注意事項</u>
3GFARMMON のコマンドおよび弊社提供のソフトウェア更新関数を使わずに、ソフトウェア更新機能を実装する場合は、株式会社エヌ・ティ・ティ・ドコモでの動作確認試験が必要となります。詳細はドコモ法人営業担当者までお問い合わせください。

6.3.2 APN の設定

UM01-HW で PPP 接続するためには APN を設定する必要があります。APN の設定に関する AT コマンドの詳細は、FOMA ユビキタスモジュール(UM01-HW)取扱説明書をご参照ください。手動での設定方法については、3G-FARM スタートアップマニュアル 6 章をご参照ください。

7 ライブラリ

PIC で動作するアプリケーションを作成する際に利用可能な UM01-HW と RTC、Armadillo-420 制御用の関数を用意しています。この章ではそれぞれの関数の仕様を説明します。

7.1 ライブラリの使用方法

MPLAB IDE で 3G-FARM 用ライブラリを使用する場合は、マニュアルディスクに収録されている 3GFARM.A をプロジェクトに追加してください。アプリケーションからライブラリの関数を利用する場合は、使用する関数のヘッダファイルをアプリケーションプログラムのソースでインクルードしてください。

7.2 UM01-HW の電源 ON/OFF

7.2.1 um01hw_powerOn

#include "um01hwpower.h"

フォーマット	int um01hw_powerOn(void);
引数	なし
関数の戻り値	int 成功した場合 0，失敗した場合 0 以外
動作	UM01-HW の電源投入シーケンスを実行します。

7.2.2 um01hw_powerOff

#include "um01hwpower.h"

フォーマット	int um01hw_powerOff(void);
引数	なし
関数の戻り値	int 成功した場合 0，失敗した場合 0 以外
動作	UM01-HW の電源切断シーケンスを実行します。

7.3 A420 の電源 ON/OFF

7.3.1 armadillo_powerOn

#include "a420power.h"

フォーマット	void armadillo_powerOn(void);
引数	なし
関数の戻り値	なし
動作	A420 の電源を投入します。

7.3.2 armadillo_powerOff

#include “a420power.h”
フォーマット　void armadillo_powerOff(void);
引数　なし
関数の戻り値　なし
動作　A420 の電源を切断します。

7.4 PIC UART 制御

uart2.h には UART のピン割り当て設定も含んでいます。

7.4.1 uart2_start

#include “uart2.h”
フォーマット　void uart2_start(int bgr);
引数　ボーレートレジスタ設定値
関数の戻り値　なし
動作　UM01-HW 用に UART2 を初期化し、オープンします。
受信バッファサイズは 500 バイトに設定しています。
bgr で指定された値をボーレートレジスタに設定します。
ボーレートレジスタ設定値の計算方法は以下の通りです。
FCY = 周波数(7.3728MHz) ×4(PLL 使用時) / 2
bgr = FCY / (4 * ボーレート) -1
ボーレート以外の通信条件は 8bit、パリティ無、1 ストップビットです。

7.4.2 uart2_close

#include “uart2.h”
フォーマット　void uart2_close(void);
引数　なし
関数の戻り値　なし
動作　UM01-HW 用 UART をクローズします。

7.4.3 uart2_char_avail

#include “uart2.h”
フォーマット　int uart2_char_avail(void);
引数　なし
関数の戻り値　受信バイト数
動作　UM01-HW から受信したバイト数を取得します。

7.4.4 uart2_getchar

#include “uart2.h”
フォーマット　char uart2_getchar(void);
引数　なし
関数の戻り値　1 バイト文字
動作　UM01-HW から受信したデータを 1 バイト取得します。

7.4.5 uart2_putchar

#include “uart2.h”
フォーマット　void uart2_putchar(char c);
引数　1 バイト文字
関数の戻り値　なし
動作　UM01-HW へ 1 バイト送信します。

7.4.6 uart2_putstr

#include “uart2.h”
フォーマット　void uart2_putstr(char *s);
引数　文字列の先頭ポインタ
関数の戻り値　なし
動作　UM01-HW へ文字列を終端まで送信します。
NULL 文字を終端として扱います。

7.5 UM01-HW ソフトウェア更新機能

7.5.1 um01hw_softwareUpdate

#include “swupdate.h”
フォーマット　int um01hw_softwareUpdate(void);
引数　なし
関数の戻り値　int

0	成功
-1	ソフトウェア更新失敗
20	ソフトウェア更新失敗もしくは中断
40	最新ソフトウェアのためソフトウェア更新不要
50	ソフトウェア更新のダウンロードに時間がかかりすぎて中止
126	電波の状態が良くありません。設置場所を見直してください。

動作　AT コマンド型ソフトウェア更新を行います。
ソフトウェア更新失敗(-1)の場合には復旧フローを行ってください。この関数の実行前に UM01-HW の電源を ON し、uart2_start()で UART2 が使用できる状態にしてください。

7.6 UM01-HW 復旧フロー

7.6.1 um01hw_recoveryFlow

#include “recovery.h”

フォーマット　int um01hw_recoveryFlow(void);

引数　なし

関数の戻り値　int 成功した場合は 1，失敗した場合には-1

動作　UM01-HW の不安定な状態の復旧を行います。条件は FOMA ユビキタスモジュール(FOMA UM01-HW)取扱説明書、組込みガイドラインをご参照ください。この関数の実行前に UM01-HW の電源を ON し、uart2_start()で UART2 が使用できる状態にしてください。

7.7 時刻取得・設定

7.7.1 open_i2c

#include “rtc.h”

フォーマット　void open_i2c(int brg);

引数　I^2C ボーレートレジスタ設定値

関数の戻り値　なし

動作　I^2C バスを初期化し、オープンします。
指定されたボーレート値を I^2C ボーレートレジスタに設定します。
I^2C ボーレートレジスタ設定値の計算方法は以下の通りです。
FCY[Hz] = 周波数(7.3728MHz) ×4(PLL 使用時) / 2
bgr = (FCY[Hz] / (2 * I^2C クロックレート[Hz])) -1

7.7.2 close_i2c

#include “rtc.h”

フォーマット　void close_i2c(void);

引数　なし

関数の戻り値　なし

動作　I^2C をクローズします。

7.7.3 rtc_init

#include “rtc.h”

フォーマット　int rtc_init(void);

引数　なし

関数の戻り値　int 初期化が実施されなかった(正常計時中)の場合:0
初期化に成功した場合:1
初期化に失敗した場合:-1

動作　RTC を初期化します。日付や時刻はすべて 0 に設定されます。RTC が正常に刻時している場合は、この関数を実行しても初期化されません。

7.7.4 rtc_gettime

#include “rtc.h”

フォーマット	void rtc_gettime(struct tm *time);
引数	struct tm *time <time.h>で定義される tm 構造体へのポインタ
関数の戻り値	なし
動作	tm 構造体ポインタで指定された領域に日付時刻を取得します。 なお、tm.yday には 1 月 1 日からの日付ではなく、0 が入ります。

7.7.5 rtc_settime

#include “rtc.h”

フォーマット	void rtc_settime(struct tm *time);
引数	struct tm *time <time.h>で定義される tm 構造体へのポインタ
関数の戻り値	なし
動作	引数の tm 構造体ポインタで指定された日付時刻を RTC に設定します。 なお、tm.yday に設定された値は使用されません。

7.7.6 rtc_setalarm

#include “rtc.h”

フォーマット	void rtc_setalarm(struct tm *time);
引数	struct tm *time <time.h>で定義される tm 構造体へのポインタ
関数の戻り値	なし
動作	引数で指定された tm 構造体ポインタで指定された時刻をアラーム時刻として RTC に設定します。アラームで指定可能なのは時(tm.hour)と分(tm.min)のみです。この関数を実行すると、指定された時刻に RTC の割り込みが発生します。

7.7.7 rtc_getalarm

#include “rtc.h”

フォーマット	void rtc_getalarm(struct tm *time);
引数	struct tm *time <time.h>で定義される tm 構造体へのポインタ
関数の戻り値	なし
動作	tm 構造体ポインタで指定された領域に次回 RTC アラームの時刻を取得します。tm.hour と tm.min にのみデータが入ります。その他のメンバの値は無効です。 rtc_setalarm()を実行していない場合、取得した値は無効です。

7.7.8 rtc_clearalarm

#include “rtc.h”
フォーマット　void rtc_clearalarm(void);
引数　なし
関数の戻り値　なし
動作　アラーム割り込みレジスタをクリアします。この関数はアラーム割り込み処理で実行してください。

7.7.9 rtc_disablealarm

#include “rtc.h”
フォーマット　void rtc_disablealarm(void);
引数　なし
関数の戻り値　なし
動作　設定されたアラームの発生を禁止します。

7.8 ウェイト

7.8.1 initTimer

#include “l_timer.h”
フォーマット　void initTimer(int pr);
引数　1ms タイマレジスタ設定値
関数の戻り値　なし
動作　1ms タイマの初期化を行います。PIC の動作周波数に合わせて 1ms タイマレジスタ設定値を設定してください。1ms タイマレジスタ設定値の計算方法は以下の通りです。
FCY = 周波数(7.3728MHz) ×4(PLL 使用時) / 2
pr = 1000us /(1 / (FCY /1000000) * 256) -1

7.8.2 delay_1ms

#include “l_timer.h”
フォーマット　void delay_1ms(unsigned int t);
引数　unsigned int t ウェイトする時間(ms)
関数の戻り値　なし
動作　t で指定された時間待ちをします。

8 3GFARMMON の機能

3GFARMMON は 3G-FARM の PIC に書込されている標準ファームウェアです。3G-FARM の CN1 (PIC の UART1)または A420 の ttymxc4 から UM01-HW の制御や PIC の入出力、割込設定、A420 の電源制御をコマンドで操作することができる機能(コマンドモード)と、PPP 通信のために UM01-HW の入出力を ttymxc2 に中継する機能(中継モード)が用意されており、CON9_1 から出力するレベルで切り替えることができます。

8.1 コマンド用シリアルチャンネル切替

3GFARMMON のコマンドモード入出力に使用するシリアルポートは、JP4 で選択することができます。JP4 で選択されるシリアルチャンネルを表 8-1に示します。

表 8-1 JP4 によるコマンドモード用シリアルチャンネル切替

ジャンパ	設定	シリアルチャンネル
JP4	1-2	ttymxc4
	2-3	CN1

8.2 GPIO9_1 による動作モード切替

ttymxc4 をコマンドモード入出力に使用する場合、GPIO9_1 に 1 を出力する必要があります。GPIO9_1 に 0 を出力すると 3GFARMMON は中継モードとなり、UM01-HW の入出力は ttymxc2 に中継され、ttymxc4 からのコマンド入力は受け付けない状態となります。JP4 でコマンドモード用シリアルチャンネルが CN1 に設定されている場合は、GPIO9_1 の出力状態に関係なくコマンドモードで動作します。

8.3 通信仕様

CN1・ttymxc4 の通信仕様は次のとおりです。

通信速度	115200
データビット長	8
ストップビット長	1 または 2
パリティビット	なし
ローカルエコー	なし
フロー制御	なし

ttymxc2 の通信仕様は次のとおりです。

通信速度	115200
データビット長	8
ストップビット長	1
パリティビット	なし
ローカルエコー	なし
フロー制御	ハードウェア

8.4 起動

3GFARMMON は 3G-FARM に電源が投入されると起動します。

8.5 コマンドモード

UM01-HW や PIC、A420 の電源操作、スリープなどの操作を行うコマンドが用意されています。コマンド一覧を表 8-2に示します。

表 8-2 コマンドモードのコマンド

コマンド名	機能
UON	UM01-HW を電源 ON します。
UOFF	UM01-HW を電源 OFF します。
SWUP	UM01-HW のソフトウェア更新を行います。
RF	UM01-HW の復旧フローを行います。
AON	A420 を電源 ON します。
AOFF	A420 を電源 OFF します。
US	UM01-HW をスリープ状態へ移行します。
UW	UM01-HW をウェークアップ状態へ移行します。
PS	PIC をスリープ状態へ移行します。
INT	割込発生した要因を応答します。
I	メモリ内容をダンプ表示します。
O	メモリ内容を変更します。
IO	CN5,9 の状態を表示、変更します。
EI	CN5,9 の指定したピンの割込を設定します。
DI	CN5,9 の指定したピンの割込を解除します。
ALM	アラーム割り込みを設定・確認・解除します。
UM	UM01-HW に直接 AT コマンドを送信します。

【数値データタイプ】
I|O コマンドでは引数あるいは設定値として入力する数値には、次の 3 種類の表現を用いることができます。

- 16 進数
  通常、入力された数値は全て 16 進数として解釈されます。
  また、数値の前に$をつけた場合も 16 進数として扱われます。
  【例】 3FFF
  　　　$2F
- 10 進数
  数値の前に#をつけた場合には、それに続く数値は 10 進数として扱われます。数値は整数のみ有効で、±符号はつけることができません。
  【例】 #25
  　　　#3500
- 2 進数
  数値の前に@をつけた場合には、それに続く数値は 2 進数として扱われます。
  【例】 @10101111
  　　　@0100

8.5.1 UON(UM01-HW 電源 ON)

【フォーマット】
UON
【応答コード】
OK　　コマンド正常終了
ERROR コマンド異常終了
【機能】

UM01-HW の電源を ON します。UM01-HW の電源 ON に成功した場合には「OK」失敗した場合には「ERROR」が返ります。「ERROR」の場合には RF コマンドで復旧フローを実行してください。UM01-HW の起動状態確認を行うため、応答コードが返るまで 30 秒から数分程度かかることがあります。このコマンドを実行すると LED3 が点滅をはじめ、完了すると LED3 が点灯状態となります。LED3 が点滅・点灯中に 3G-FARM の電源を切断すると、UM01-HW が故障する可能性がありますのでご注意ください。

8.5.2 UOFF(UM01-HW 電源 OFF)

【フォーマット】
UOFF
【応答コード】
OK　　コマンド正常終了
ERROR コマンド異常終了
【機能】

UM01-HW の電源を OFF します。UM01-HW の電源 OFF に成功した場合は「OK」、失敗した場合には「ERROR」が返ります。「ERROR」の場合には RF コマンドで復旧フローを実行してください。

8.5.3 SWUP(ソフトウェア更新実行)

【フォーマット】
SWUP
【機能】

UM01-HW のソフトウェア更新を行います。ソフトウェア更新の応答が「ERROR」の場合 RF コマンドで復旧フローを実行してください。詳細は FOMA ユビキタスモジュール(FOMA UM01-HW)取扱説明書、組込ガイドラインをご参照ください。

表 8-3 ソフトウェア更新応答コード

応答コード	内容
OK	ソフトウェア更新正常終了
ERROR	UM01-HW が異常状態です。復旧フローを行ってください。
ERROR00	電波の状態が良くありません。設置場所を見直してください。
ERROR20	ソフトウェア更新に失敗もしくは中断しました。
ERROR40	最新のソフトウェアです。

8.5.4 RF(UM01-HW 復旧フロー)

【フォーマット】
RF
【応答コード】
OK　　コマンド正常終了
ERROR コマンド異常終了
【機能】

UM01-HW の復旧フローを行います。詳細は FOMA ユビキタスモジュール(FOMA UM01-HW)取扱説明書、組込ガイドラインをご参照ください。
成功すると「OK」と応答します。失敗すると「ERROR」と応答し UM01-HW の電源を OFF します。

8.5.5 AON(Armadillo-420 電源 ON)

【フォーマット】
AON
【応答コード】
OK　　コマンド正常終了
【機能】

Armadillo-420 の電源を ON します。

8.5.6 AOFF(Armadillo-420 電源 OFF)

【フォーマット】
AOFF ＜終了待機時間(秒)＞
【応答コード】
OK　　コマンド正常終了
【機能】

指定された終了待機時間後、Armadillo-420 の電源を OFF します。設定する終了待機時間は Armadillo-420 に接続されている機器等の電源切断準備にかかる時間を考慮して適切な時間を設定してください。＜終了待機時間＞を省略すると、待機時間をとらずに Armadillo-420 の電源を切断します。このコマンドを ttymxc4 から実行した場合は、PIC も同時にスリープ状態となります。

8.5.7 US(UM01-HW スリープ状態移行)

【フォーマット】
US
【応答コード】
OK　　コマンド正常終了
ERROR コマンド異常終了
【機能】

UM01-HW をスリープ状態に移行します。この状態で音声着信/パケット着信/SMS 着信があると、PIC が UM01-HW をアイドル状態に戻し、A420 が電源 OFF の場合は、電源を供給して起動します。A420 で起動後に動作させるアプリケーションで INT コマンドを使用し、起動要因を確認することができます。

8.5.8 UW(UM01-HW ウェークアップ状態移行)

【フォーマット】
UW
【応答コード】
OK　　コマンド正常終了
ERROR コマンド異常終了
【機能】
　UM01-HW をウェークアップし、アイドル状態に移行します。

8.5.9 PS(PIC スリープ状態移行)

【フォーマット】
PS
【機能】
　PIC をスリープ状態に移行します。
　RTC,UM01-HW,PORT の割込がが発生するまで SLEEP 状態を保持します。各割込が発生すると PIC はウェークアップ状態になります。同時に A420 が電源 OFF 状態の場合には A420 を起動します。A420 は起動の際、起動要因を確認してください。起動要因は INT コマンドの応答で確認できます。

8.5.10 INT (割込要因問い合わせ)

【フォーマット】
INT
【機能】
　割込発生要因を応答します。割込要因は表 8-4をご参照ください。応答コードは数字文字列８桁です。複数の割込み要因がある場合は、それぞれの応答コードの数値が加算された値が応答となります。割込要因問い合わせによって、割込みに対する応答コードはクリアされます。

表 8-4 割込要因応答コード

応答コード	割込要因
00000000	割込要因なし
00000001	RTC アラーム
00000002	UM01-HW SMS/音声/パケット着信
00010000	ポート CN5-1
00020000	ポート CN5-2
00040000	ポート CN5-3
00080000	ポート CN5-4
00100000	ポート CN5-5
00200000	ポート CN5-6

応答コード	割込要因
00400000	ポート CN5-7
00800000	ポート CN5-8
01000000	ポート CN7-7
02000000	ポート CN7-8
04000000	ポート CN7-9
08000000	ポート CN7-10
10000000	ポート CN7-11
20000000	ポート CN7-12
40000000	ポート CN7-13
80000000	ポート CN7-14

8.5.11 I(ポート入力)

【フォーマット】
I ＜ポートアドレス＞ [/C]
【機能】
〈ポートアドレス〉で指定された PIC の入力ポートからデータを読み、値を 16 進・2 進の文字列で返します。/C オプションがつけられている場合は、使用しているシリアルチャンネルから 1 文字送信されるまでこの動作を繰り返します。この機能はハードウェアのデバック時に便利です。

8.5.12 O(ポート出力)

【フォーマット】
O ＜ポートアドレス＞＜出力データリスト＞ [/C]
【機能】
PIC の〈ポートアドレス〉で指定した出力ポートに、〈出力データリスト〉で与えられたデータを書き込みます。/C オプションがつけられている場合は、使用しているシリアルチャンネルから 1 文字送信されるまでこの動作を繰り返します。この機能はハードウェアデバッグ時に便利です。
〈出力データリスト〉には､複数個のデータをスペースまたはカンマで区切って一度に与えることができます。/C オプションが指定されている場合はこれらのデータが繰り返し使用されます。

8.5.13 IO(GPIO 入力・出力)

【フォーマット】
IO ＜CN 番号-ピン番号＞＜データ指定文字＞
【機能】
3G-FARM の GPIO に対してデータ入出力を行います。GPIO ピンは〈CN 番号-ピン番号〉の形で指定します。例えば CN9-7 端子の場合、CN97 または CN9-7、CN9.7 などと指定することができます｡〈出力データ〉に指定可能な文字と動作は表 8-5をご参照ください。〈出力データ〉を指定しない場合には指定したピンのデータを読み、0 または 1 を返します。

表 8-5 データ指定文字

データ指定文字	動作
0	0 を出力
1	1 を出力
U	プルアップ入力に設定
D	プルダウン入力に設定
Z	プルアップ・ダウンのない入力に設定

8.5.14 EI(割込設定)

【フォーマット】
EI ＜CN 番号-ピン番号＞
【応答コード】
OK　　コマンド正常終了
ERROR コマンド異常終了
【機能】

3G-FARM の GPIO 端子からの割込みを設定します。GPIO ピンは<CN 番号-ピン番号>の形で指定します。例えば CN9-7 端子の場合、CN97 または CN9-7、CN9.7 などと指定することができます。指定された端子に状態変化が発生すると、割込みが発生します。PS コマンドで PIC がスリープしている場合、この GPIO 割込みでウェークアップすることができます。指定できない端子を指定した場合は、ERROR 応答となります。

8.5.15 DI(割込解除)

【フォーマット】
DI ＜CN 番号-ピン番号＞
【応答コード】
OK　　コマンド正常終了
ERROR コマンド異常終了
【機能】

3G-FARM の GPIO 端子からの割込みを解除します。GPIO ピンは<CN 番号-ピン番号>の形で指定します。例えば CN9-7 端子の場合、CN97 または CN9-7、CN9.7 などと指定することができます。指定できない端子を指定した場合は、ERROR 応答となります。

8.5.16 ALM(RTC アラーム設定)

【フォーマット】
ALM [＜時＞＜分＞] | [D]
【応答コード】
OK　　コマンド正常終了
【機能】

引数なしでコマンドを実行すると、RTC からアラーム時刻を読み出して設定時刻を時 2 桁＋コロン(:)+分 2 桁で応答します。時刻パラメータをつけて実行した場合には、指定されたアラーム時刻を設定します。アラーム時刻(時・分)は 10 進数で指定し、時・分の間はスペースで区切ってください。時刻は 24 時間制で設定します。設定値の正当性はチェックしていませんので、不正な値を設定しないでください。
RTC は PIC と Armadillo-420 で共有しています。Armadillo-420 が RTC に設定する時刻は UTC ですのでご注意ください。
設定されたアラーム時刻になると、RTC 割込みが発生します。PS コマンドで PIC がスリープしている場合、この割込みでウェークアップすることができます。
コマンドの引数に D を指定した場合は、設定されているアラームを解除します。

8.5.17 UM(AT コマンド直接送受信)

【フォーマット】
UM
【機能】
このコマンドを実行すると UM01-HW と直接 AT コマンドの送受信ができる状態となります。AT コマンドの仕様の詳細は FOMA ユビキタスモジュール(FOMA UM01-HW)取扱説明書、組込ガイドラインをご参照ください。[ESC]キー入力でこのコマンドを終了し、プロンプトに戻ります。
なお送信データがエコーバックされるかどうかは UM01-HW の設定に依存します。

9 サンプルスクリプト

3GFARMMON を A420 から使用するサンプルとして、簡単なシェルスクリプトを用意しています。3GFARMMON を A420 から使用するため、JP4 を 1-2 に設定してください。

9.1 UM01-HW 電源 ON

シェルスクリプト：uon.sh
UON コマンドを実行します。LED3 が点滅中は UM01-HW 電源 ON 処理中です。LED3 が点灯すると UM01-HW は使用可能です。

9.2 UM01-HW 電源 OFF

シェルスクリプト：uoff.sh
UOFF コマンドを実行します。LED3 が点滅中は UM01-HW 電源 OFF 処理中です。LED3 が消灯すると処理完了です。LED1、2 も消灯します。

9.3 UM01-HW スリープ状態移行

シェルスクリプト：us.sh
US コマンドを実行します。UM01-HW をスリープ状態へ移行します。LED3 は点灯のまま、LED1、2 が消灯します。音声着信/パケット着信/SMS 着信でウェークアップします。

9.4 UM01-HW ウェークアップ

シェルスクリプト：uw.sh
UW コマンドを実行します。UM01-HW をウェークアップし、アイドル状態へ移行します。LED3 は点灯のままです。LED1 が消灯から点滅に変わります。

9.5 PIC スリープ状態移行

シェルスクリプト：ps.sh
PS コマンドを実行し、PIC をスリープ状態へ移行します。LED3 は点灯のままです。設定した割込もしくは音声着信/パケット着信/SMS 着信で復帰します。

9.6 A420 電源 OFF

シェルスクリプト：aoff.sh
A420 の電源を OFF します。設定した割込もしくは音声着信/パケット着信/SMS 着信で A420 を起動します。3G-FARM の電源 OFF、ON でも復帰しますが UM01-HW の電源 ON 中は行わないでください。

9.7 RTC アラーム設定

シェルスクリプト：alm.sh
起動すると A420 を起動する時刻を入力します。その後 A420 をオフし、指定された時刻に A420 が起動します。

9.8 PPP 接続、web 公開

シェルスクリプト：pppon.sh
PPP 接続して web 公開します。ifconfig コマンドで確認した IP アドレスにアクセスすると、web ページを確認できます。
http://(IP アドレス)/index.cgi
事前に A420 の PPP 接続に関する設定と UM01-HW の APN 設定を行ってください。設定方法はスタートアップマニュアルをご参照ください。

9.9 PPP 切断

シェルスクリプト：pppoff.sh
PPP 切断します。LED1 が点灯から点滅に変わります。

3G-FARM ソフトウェアマニュアル　　2013 年 6 月 21 日 rev.1.2

梅澤無線電機 株式会社

http://www.umezawa.co.jp/

納期・価格等のお問い合わせは

新事業推進室 FreeDial ０１２０－００２２０５

製品・資料についての技術的なお問い合わせは

技術推進部 FreeDial ０１２０－０２４７６８